WEST

Generate Collection Print

L9: Entry 1 of 5 File: USPT Jan 29, 2002

DOCUMENT-IDENTIFIER: US 6342892 B1
TITLE: Video game system and coprocessor for video game system

Brief Summary Paragraph Right (4):
3D graphics are fundamentally different from 2D graphics. In 3D graphics techniques, a "world" is represented in three dimensional space. The system can allow the user to select a viewpoint within the world. The system creates an image by "projecting" the world based on the selected viewpoint. The result is a true three-dimensional image having depth and realism.

Drawing Description Paragraph Right (18):
FIG. 8 shows an example hierarchical task list including graphics display lists and audio play lists;

Drawing Description Paragraph Right (28):
FIG. 15 shows an example simple signal processor play list processing example;

Detailed Description Paragraph Right (23):
The list of graphics commands is called a "display list" because it controls the images coprocessor 200 displays on the TV screen 60. The list of audio commands is called a "play list" because it controls the sounds that are played over loudspeaker 62. Generally, main processor 100 specifies both a new display list and a new play list for each video "frame" time of color television set 58.

Detailed Description Paragraph Right (84):
FIG. 8 shows an example of a task list 250. The task list 250 may reference one or more display lists and/or play lists 110. These display lists or play lists 110, in turn, may reference additional data structures including other display lists or play lists. A display list 110 can point to other display lists and/or graphics data. Similarly, a play list can reference other play list and/or sound data. In this example, display lists and play lists can be thought of as hierarchical data structures up to ten levels deep. Signal processor 400 processes the display lists and play lists of the stack, pushing and popping the current display list pointer. All display lists must terminate with an "send" command. For example, display list 110(1) shown in FIG. 8 references another display list 110(2). Display list 110(2) references graphics data 112 needed to execute the list. Similarly, play list 110(4) shown in FIG. 8 references sound data 112B.

Detailed Description Paragraph Right (87):

Now that signal processor 400 has a task list and is started, it proceeds to perform each of the operations requested in the task list. It continues to execute the task list until it reaches the end of the task list, at which time it stops and waits for main processor 100 to provide a new task list. Generally, main processor 100 provides a new task list once each video frame--although, as discussed above, in many cases only a portion of the task list and/or the display and/or play lists the task list references may actually change from one frame to the next. Portions of the task list in main memory 300 may be "double buffered" so the main processor 100 can be writing to one buffer while signal processor 400 reads from another buffer. Before the next video frame, the main processor 100 can change a pointer to give the signal processor 400 access to the new buffer.

Detailed Description Paragraph Right (88):
As signal processor 400 executes the task list, it retrieves additional SP microcode 156 modules from main memory 300 as needed to perform the specified tasks. For example, signal processor 400 may use its DMA facility 454 to load particular graphics microcode into instruction memory 402 to execute graphics commands specified by a task list, and may similarly retrieve and load audio processing microcode routines to perform audio processing specified by the task list. Different microcode routines or "overlays" may be loaded on an as-needed basis to more optimally handle particular types of graphics and/or audio processing operations. As one example, the signal processor 400 may load special lighting graphics routines as overlays to perform particular lighting operations, and may load clipping routines or overlays to perform particular culling operations. Microcode loading and reloading into signal processor 400 during execution of the single task list 250 is necessary in this example because signal processor instruction memory 402 is not large enough to store all of SP microcode 156, and the signal processor is designed so that it can execute instructions only out of its internal instruction memory.

Detailed Description Paragraph Right (145):
Signal processor 400 next performs lighting calculations in order to "light" each of the vertices specified in the vertex command. System 50 supports a number of sophisticated real-time lighting effects, including ambient (uniform) lighting, diffuse (directional) lights, and specular highlights (using texture mapping). In order to perform lighting calculations in this example, signal processor 400 must first load an SP microcode 108 overlay to perform the lighting calculations. The G_SETGEOMETRYMODE command must have specified that lighting calculations are enabled, and the lights must have been defined by the G_NUM_LIGHTS command discussed above. The part of microcode 108 that performs the lighting calculations is not normally resident within signal processor 400, but is brought in through an overlay when lighting calls are made. This has performance implications for rendering scenes with some objects lighted and others colored statically. In this example, the lighting overlay overwrites the clipping microcode, so to achieve highest performance it is best to minimize or completely avoid clipped objects in lighted scenes.

Detailed Description Paragraph Right (187):
In this example, signal processor 400 is shared between graphics processing and digital audio signal processing. Because of the high

speed calculating capabilities of signal processor vector unit 420, signal processor 400 is able to complete processing of the audio to be played during the next succeeding video frame in much less than the current video frame time, and is also able to complete graphics processing for the image to be displayed during the next succeeding image in less than the current frame time. This allows task list 250 to specify both graphics display lists and audio play lists that all must be completed by signal processor 400/coprocessor 200 by the beginning of the next video frame time. However, in this example there is nothing to prevent main processor 100 from giving coprocessor 200 a task list 250 that the coprocessor cannot complete before the next video frame begins. If the combined audio and graphics processing required by signal processor 400 is sufficiently intensive and time-consuming, the signal processor 400 can work on processing the task list for the entire current video frame time and still not be done by the beginning of the next video frame. It is up to video game program 108 to avoid overtaxing coprocessor 200, and to handle any overtaxing in an appropriate manner should it occur. A video game programmer can avoid overtaxing signal processor 400 by ensuring that all display lists 110 are organized efficiently, modeling the objects in 3-D in an efficient manner, and taking precautions to ensure that extensive time consuming processing (e.g., clipping) is avoided or minimized. Even with such precautions, however, it may take coprocessor 200 more than a single video frame time to complete especially complicated images. A video game programmer can handle this situation by slowing down the effective frame rate so that television 58 redisplays the same image stored in one part of frame buffer 118 for multiple video frames during which time coprocessor 200 can complete processing the next image. Because the user may perceive a variable frame rate as undesired delay, it is often best to slow down the overall effective frame rate to the rate required for coprocessor 200 to complete the most processing-intensive images--thus preventing more complex images from appearing more slowly than less complex images.

Detailed Description Paragraph Right (190):
Sequence player 702, sound player 704 and synthesis driver 706 may differ depending on the particular video game being played. In general, sequence player 702 is responsible for the playback of Type 0 MIDI music sequence files. It handles sequence, instrument bank and synthesizer resource allocation, sequence interpretation, and MIDI message scheduling. Sound player 704 is responsible for the playback of all ADPCM compressed audio samples. It is useful for sound effects and other streamed audio. Synthesis driver 706 is responsible for creating audio play lists 110 which are packaged into tasks by main processor 100 under software control and passed to coprocessor 200 in the form of task lists 250. In this example, synthesis driver 706 allows sound player 704 or other "clients" to assign wave tables to synthesizer voices, and to control playback parameters. As discussed above, the audio synthesis microcode 708 processes tasks passed to it and synthesizes L/R stereo 16-bit samples, which signal processor 400 deposits into audio buffers 114 within main memory 300 for playback via audio interface 208, audio DAC 140 and amplifier/mixer 142.

Detailed Description Paragraph Right (192):
FIG. 15 shows an example of a simple signal processor play list process. The FIG. 15 process is specified by a play list 110 generated

by main processor 100 under control of video game program 108, and specified as part of a task list 250. Thus, the FIG. 15 SP play list process is an example of an output of synthesis driver 706 that is provided to signal processor 400 in the form of an audio play list 110.

Detailed Description Paragraph Right (193):
Because of the limited size of instruction memory 402 in this example, audio synthesis microcode 708 is generally not continuously resident within signal processor 400. Instead, the initialization microcode main processor 100 arranges to be loaded into instruction memory 402 (see FIG. 9, block 604), ensures that the appropriate audio microcode routine is loaded into the instruction memory for audio processing (also ensures that the appropriate graphics microcode routine is loaded into the instruction memory for graphics processing). The steps shown in FIG. 15 assume that the audio synthesis microcode 708 is resident within the signal processor instruction memory 402, and that the signal processor 400 is reading an audio play list 110 specifying the steps shown.

Detailed Description Paragraph Right (194):
Generally, the first task of an audio play list 110 is to set up buffers within signal processor data memory 408 required to perform the audio processing task (FIG. 15, block 710). Generally, this buffer set up process involves allocating areas within data memory 404 to be used as one or more audio input buffers, and allocating an audio output buffer within the data memory. Generally, main processor 100 also commands signal processor 400 to use its DMA facility 454 to retrieve audio input data 112b from main memory into the allocated input buffer(s) for processing. Main processor 100 may next set certain attributes (e.g., volume ranges and change rates) to be used for the audio processing (FIG. 15, block 712). Main processor 100 then specifies the types of signal processing to be performed by signal processor 400 along with appropriate parameters (FIG. 15, block 714). In this example, main processor 100 can specify decompression, resampling, envelope/pan, mixing, and other processing (e.g., reverb) to be performed individually or in combination. The audio play list 110 typically will terminate with a command to save the contents of the output audio buffer stored in signal processor data memory 404 into main memory 300 (block 716).

Detailed Description Paragraph Right (195):
FIG. 16 shows the overall tasks performed by audio synthesis microcode 708 in this example. Signal processor 400 under microcode control retrieves the next play list command from the current audio play list 110, and determines what kind of command it is (FIG. 16, block 718). In this example, the audio command within an audio play list 110 may fall into the following general types:

Detailed Description Paragraph Right (198):
If the next play list command is a flow control command, signal processor 400 responds to the command by traversing the current audio play list in the manner specified by the command. Nesting of audio play lists 110 is preferably permitted, and signal processor 400 may maintain an audio play list stack in main memory 300 (just as it may do for graphics display lists).

Detailed Description Paragraph Right (199):
If the next audio play list command is an attribute command, signal processor 400 processes the command by establishing appropriate mode and/or attribute conditions to be used for subsequent audio processing (FIG. 16, block 724). In this example, audio synthesis microcode 708 supports the following example attribute command format and associated function:

Detailed Description Paragraph Right (200):
If the next audio play list command retrieved by signal processor 400 is a decompression command, the signal processor performs a decompression operation to decompress a compressed audio binary stream stored in an input buffer within data memory 404 to produce 16-bit audio samples which it stores in a defined audio output buffer within its data memory (FIG. 16, block 726). In this example, audio synthesis microcode 708 supports the following audio decompression command format and associated function:

Detailed Description Paragraph Right (202):
If the next audio play list command signal processor 400 reads is a resample command, then the signal processor provides pitch shifting/resampling as well as integral envelope modulation based on the parameters specified in the command (FIG. 16, block 728). The following is an example of a resample command and associated function supported by audio synthesis microcode 708.

Detailed Description Paragraph Right (204):
If the next audio play list command signal processor 400 reads is an envelope/pan command, the signal processor performs that command by modulating one or two audio signal streams using a linear envelope (FIG. 16, block 730). An envelope command multiplies an audio input sample stream by a linear function, and is thus able to ramp the volume of the audio up or down. A "pan" command generally applies inverse linear functions to audio in left and right stereo channels--accomplishing the effect of moving the perceived source of a sound or voice in space (i.e., from left to right or from right to left). The following examples of envelope/pan command formats and associated functions are supported by audio synthesis microcode 708 in this example of system 50.

Detailed Description Paragraph Right (207):
If the next audio play list command is a mixing command, signal processor 400 performs a mixing function to mix two audio input sample streams into the output audio buffer (FIG. 16, block 732). The following example mixing command format and associated function is supported by signal processor 400 and audio synthesis microcode 708 in this example.

Detailed Description Paragraph Right (208):
If the next audio play list command is a special signal processing/effects command, signal processor 400 executes the command by providing the specified special effect or signal processing (FIG. 16, block 734). An example special signal processing/effect is the addition of reverberation to create presence. This special effect simulates sound reflection in caves, concert halls, etc., and can also be used for various other special effects. Signal processor 400 and audio synthesis microcode 708 supports the following example

reverberation special effects command format and associated function:

Detailed Description Paragraph Left (36):
Example Play List Processing

Detailed Description Paragraph Type 1 (19):
Play list audio commands 110b

Detailed Description Paragraph Type 1 (51):
Play list processing

Other Reference Publication (5):
Perry et al., associate editors, "Special Report, Consumer Electronics, Video games: the electronic big bang," IEEE Spectrum, pp. 20-32, Dec. 1982.

Other Reference Publication (16):
MIPS Microprocessor R4000 User's Manual, First Edition, by Joe Heinrich, Copyright 1993 by MIPS Technologies, Inc.

Other Reference Publication (17):
MIPS Microprocessor R4000 User's Manual, Second Edition, by Joe Heinrich, Copyright 1994 by MIPS Technologies, Inc.

CLAIMS:

17. A system as in claim 5 wherein the vector processing unit comprises plural calculating units and the signal processor includes a data memory for storing data words and a multiplexer coupled between the vector processing unit and the data memory, the multiplexer steering data words from the memory into selected ones of the plural calculating units.

35. A coprocessor as in claim 21, wherein said vector processing unit comprises plural calculating units and said signal processor further comprises a data memory for storing data words and a multiplexer coupled between said vector processing unit and said data memory, said multiplexer steering data words from said memory into selected ones of said plural calculating units.

WEST

Generate Collection Print

L10: Entry 9 of 21 File: USPT Jun 29, 1999

DOCUMENT-IDENTIFIER: US 5918213 A
TITLE: System and method for automated remote previewing and purchasing of music, video, software, and other multimedia products

Brief Summary Paragraph Right (17):
Other retail stores set up listening booths, in-store juke boxes, sample computers, or other like trial stations where customers can sample some of the available products. However, due to the costs of installing and maintaining these trial stations, the floor space they consume, and the cost of having inventory available at these stations, the variety of products available for sampling is usually quite limited.

Brief Summary Paragraph Right (28):
For example, when the customer accesses the automated purchasing system, he or she may be provided with the option of hearing the top hits in his or her area, the tracks featured in a local radio station's play list, a featured artist, and the like. This allows the customer to gain exposure to various products even though he or she may not be sure in which particular products he or she is interested at the time he or she accesses the system. Where the product is movies, a similar scenario can be used whereby the customer can sample movies based on top box office hits, top sales, award-winning titles, films featuring certain stars, and other similar criteria. This facilitates browsing among the various selections offered.

Detailed Description Paragraph Right (14):
As stated above, in one embodiment the caller is provided the ability to listen to sample music selections by artist and title. The samples may include every track on an album, a selection of tracks from each album, or even a single track from an album. This feature is referred to as a Music Mall. In the Music Mall, callers can browse through the entire music inventory of the purchasing system. In the Music Mall, callers can choose what music they want to sample by categories such as, for example, Artist, Album Title, Catalog Number, Top-Selling Albums, and Specials. When the caller first connects with the music service, the caller is provided an audio menu by which he or she can make various selections. For example, the caller could choose to enter the Music Mall directly, where music titles can be purchased by Artist, Catalog Number, etc. Alternatively, the caller could choose to sample music featured by a local radio station, or top-selling albums in a particular style of music.

Detailed Description Paragraph Right (31):
Also included in the example architecture is an order fulfillment

center. The order fulfillment center is the center that is actually responsible for filling the customer's order and sending the order to the customer. In one embodiment, the order fulfillment center is a third-party vendor contracted to maintain an inventory of products and to fulfill customer orders based on order information provided by the interactive transaction database 112. In another example, the order fulfillment center is an in-house order processing center responsible for filling customer orders. In yet another example, the order fulfillment center is an automated order fulfillment center comprising an automated order picker to automatically retrieve ordered products from inventory to fill customer orders.

Detailed Description Paragraph Right (43):
As stated above, in step 214, the scripts played to caller 182 provide caller 182 with a menu allowing him or her to select from a number of choices. According to one embodiment, the first choice provided to caller 182 is the style of music for which he or she wishes to shop during that call. This style could be rock, jazz, pop, alternative, rap, etc. Upon selecting a style, the caller is further provided with the option to hear samples of titles from various categories in the selected style of music. The categories can include: featured artist for the local radio station, top hits from that station, the radio station play list for recently played songs, or to hear the top hits nationwide.

Detailed Description Paragraph Right (104):
As described above with reference to FIG. 1, an order fulfillment center is provided in one embodiment to fill the customer order. One example of an order fulfillment center is fulfilment vendor 436, which, according to the example architecture is separated from the rest of the automated product purchasing system via wide area network 412. When an order is completed by a customer it is sent to fulfillment vendor 436 for fulfillment of the order. Fulfillment vendor 436 receives the order via wide area network 412 and processes and ships the order. Fulfillment vendor 436 can be an actual part of the automated product purchasing system, or alternatively, fulfillment vendor 436 can be an independent service contracted to perform order fulfillment-related services such as warehousing, inventory control and shipping functions. In one embodiment, fulfillment vendor 436 is an independent contractor.

Detailed Description Paragraph Right (105):
Inventory updates to reflect new stock received and orders shipped are provided to interactive transaction database 112 via wide area network 412. In this manner, local instances of inventory information can be maintained within interactive transaction database 112, without the need for VRU site 460 to retrieve data from across the WAN 412 each time availability information is needed. As a result, when a caller 182 chooses to sample a particular title, interactive transaction database 112 can provide VRU 104 with availability information without the need to query fulfillment vendor 436 for each call. The inventory updates can be performed periodically (e.g., daily, weekly, hourly, etc.) depending on the system requirements.

Detailed Description Paragraph Right (135):
Additional scripts stored at VRU 104 are called environment scripts. These scripts are scripts that are more likely to change on a regular

basis. Examples of environment scripts include scripts provided by a local radio station disk jockey (or DJ) describing such selections as the featured artist, radio play list, and radio station top hits. Environment scripts also can include scripts announcing promotional items, scripts announcing top hits nation-wide, scripts announcing featured artists nation-wide, and so on. As these examples illustrate, environment scripts, in contrast to generic scripts, are likely to change on a regular and periodic basis to reflect current offerings of the automated product purchasing system. Like generic scripts, environment scripts are accessed by VRU 104 and played to caller 182 at the appropriate time.

Detailed Description Paragraph Right (144):
In addition to a top-hits list, featured artist lists, radio play station lists, and promotional offerings lists can be maintained in a similar fashion. The top hits list described in this embodiment is structured to operate well with the data structure illustrated in FIG. 10. It will become apparent to a person skilled in the relevant art how other list structures can be utilized to implement the described listing functionality. It is not required that a top hits list data structure 1100, as illustrated in FIG. 11, be implemented with the automated product purchasing system.

Detailed Description Paragraph Right (146):
Mapping information is used to provide information relating to geographic features of the automated product purchasing system. As described above, one feature of the automated product purchasing system is that it allows caller 182 to browse selections such as the featured artists, the top-hits list, and radio play lists for a particular radio station in the caller's geographic area. Determination of the caller's geographic area is made using mapping information stored in interactive transaction database 112.

Detailed Description Paragraph Right (159):
Profiling information and cost statistics are also maintained by interactive transaction database 112. Profiling information can include profiles on individual callers 182 and profiles on callers in general, callers in particular music categories, or callers from a particular geographic area, and so on. Profiles can include statistical information such as preferred music styles, quantity and frequency of browses, quantity and frequency of purchases, types of purchases, responses to promotional offerings, whether purchases are from the Music Mall or from other categories such as top-hits list, featured artist, play list, etc. This profiling and statistical information can be used to enhance the system functionality, and can be used by marketing professionals to improve the marketability of the products offered by the automated product purchasing system.

Detailed Description Paragraph Right (163):
Once the caller has selected a music style, the caller can choose to hear selections from a featured artist, a radio play list, top-selling albums. Alternatively, caller 182 may choose to shop by artist name or catalog identification. If the caller chooses this alternative selection, he or she enters the "Music Mall". These selections are illustrated by hierarchy level 1312. If caller 182 decides to enter the Music Mall, caller 182 may choose to shop by artist, shop by catalog number, shop based on special or promotional items, and shop

by top-selling albums. These selections are illustrated by hierarchy level 1316. Alternative embodiments may include a different menu hierarchy and different selections at each level, to tailor user shopping, to a particular product or market.

Detailed Description Paragraph Right (198):
These top-hits examples are provided in terms of a scenario where a top-five hit list for a particular station and a top-twenty nationwide hits list is sampled. Other lists such as radio station play lists (for example, a list of songs played on the associated radio station during the past half hour) other top-hits lists, promotional items lists, etc. can be sampled as well.

Detailed Description Paragraph Right (341):
In-store data 4008 also can include data pertaining to selections available at the retail store 3804. For example, in one embodiment, local data 4022 can include a price list and an inventory of all the titles available at retail store 3804. This can be provided to shopper 3808 via user interface 4004 while he or she is browsing selections either locally or via the automated product purchasing system, so that shopper 3808 knows what the items cost at the retail store and whether or not they are available in the retail store's inventory. In-store data 4008 can also include additional descriptions of in-store selections, concert information for local concerts, and other useful information.

Detailed Description Paragraph Right (370):
In a step 4412, automatic order picker 4304 retrieves their ordered items from their designated locations. As items are removed from the warehouse shelves, or after the order is filled, automatic order picker updates the inventory to reflect the fact that the item has been sold. This occurs in a step 4416. Inventory updates can be provided back to interactive transaction database 4112, such that interactive transaction database 4112 can provide availability information to the customer. Alternatively, the inventory can be maintained fully at the automatic order picker 4304 and this information queried as a user is browsing the products or when the user places an order for a product. Automatic order picker 4304 can interface with interactive transaction database 4112 via a direct connection, via a local area network or via other communications means.

Other Reference Publication (9):
Author unknown, "Voice-Response System Improves Order Entry, Inventory Control," Communication News, Aug. 1976.